Adam Ive Transformatörü

Egemen Kaya

Mendax Yazılım

x@mendaxyazilim.com

Özet

Transformatör modelleri, bir kodlayıcı ve bir kod çözücü içeren karmaşık tekrarlayan veya evrişimli sinir ağlarına dayanır. En iyi performans gösteren modeller, kodlayıcı ve kod çözücüyü bir dikkat mekanizmasıyla birbirine bağlar. Transformatör, sadece dikkat mekanizmalarına dayanarak, nüks ve kıvrımları tamamen ortadan kaldıran yeni bir basit ağ mimarisi öneriyoruz. Yapay zeka transformatörümüz, dört dakikalık midi üretim görevinde yüzde doksan iki başarıya ulaşarak, mevcutta bulunan çoğu müzik transformatörlerinden daha iyi bir performans gösterdi. Mevcut sanatçının girdileri verildiği taktirde transormatör, satırların devamını o sanatçının ölçülerine benzerlikte devam ettirerek, girdileri kopyalayarak geliştirebiliyor. Transformatör'e yüklenen girdileri kopyalayıp geliştirmesini gözlemleyerek müzik üreticilerin işlerini ne kadar kolaylaştırdığını görebiliyoruz.

1 Giriş

Tekrarlayan sinir ağları, dil modelleme ve çeviri gibi dizi modelleme ve transdüksiyon problemlerinde son teknoloji yaklaşımlar olarak sağlam bir şekilde kurulmuştur [1,2,3]. O zamandan beri, tekrarlayan dil modellerinin ve kodlayıcı-kod çözücü modellerinin sınırlarını zorlamaya yönelik çok çaba gösterilmiştir.

Tekrarlayan modeller tipik olarak giriş ve çıkış dizilerinin sembol konumları boyunca hesaplamayı etkiler. Konumları hesaplama süresindeki adımlarla hizalayarak, önceki gizli durum h_t,'in bir fonksiyonu ve konum t için giriş olarak gizli bir $h_{t\ 1}$ dizesi üretir. Bu ardaşık sistem, bellek kısıtlamaları örnekler arasında toplu işlemi sınırladığı için, daha uzun dizi uzunluklarında kritik hale gelen eğitim örnekleri içindeki paralelleştirmeyi engeller. Son zamanlarda yapılan çalışmalar, faktorizasyon hileleri ve koşullu hesaplama aracılığıyla hesaplama verimliliğinde önemli iyileşmeler sağlarken, performansı da artırmıştır.

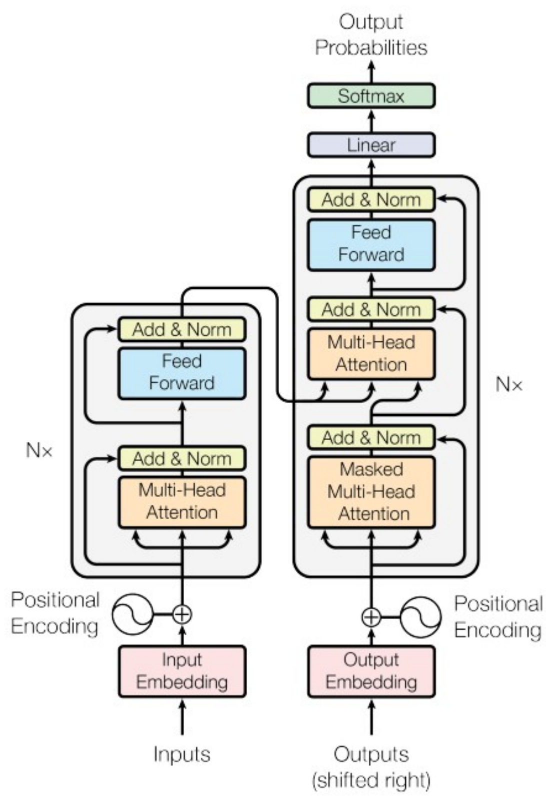
Dikkat mekanizmaları, çeşitli görevlerde zorlayıcı dizi modellemesi ve iletim modellerinin ayrılmaz bir parçası haline gelerek, girdi veya çıktı dizilerindeki mesafelerine bakılmaksızın bağımlılıkların modellenmesine izin verir. Bununla birlikte, bu tür dikkat mekanizmaları [4] tekrarlayan bir ağ ile birlikte kullanılmaktadır.

Bu çalışmada, nüksetmekten kaçınan ve bunun yerine girdi ve çıktı arasındaki küresel bağımlılıkları çekmek için tamamen bir dikkat mekanizmasına [4] dayanan bir model mimarisi olan Transformatörleri öneriyoruz. Transformatör, önemli ölçüde paralelleştirmeye izin verir ve sekiz P100 GPU'da on iki saat kadar kısa bir eğitimden sonra üretim kalitesinde iyi seviyelere ulaşabilir.

2 Transormatör Mimarisi

Transormatör modellerinin çoğu bir kodlayıcı ve kod çözücü yapısına sahiptir [3, 2, 1]. Burada kodlayıcı, sembol gösterimlerinin (x_1, ..., x_n) bir girdi dizisini sürekli z = (z_1, ..., z_n) dizilişiyle eşleştirir. Z verildiğinde, kod çözücü bir kerede bir eleman olmak üzere sembollerin bir çıkış dizisini (y_1, ..., y_m) üretir. Her adımda model otomatik gerileme özelliğine sahiptir ve daha önce üretilen sembolleri bir sonraki ürünü oluştururken ek girdi olarak tüketir.

Transformatör, bu genel mimariyi sırasıyla Şekil 1'in [4] sol ve sağ yarısında gösterilen hem kodlayıcı hem de kod çözücü için yığılmış öz dikkat ve noktasal olarak tam olarak bağlanmış katmanları kullanarak takip eder.



Şekil 1: Transformatör - model mimarisi.

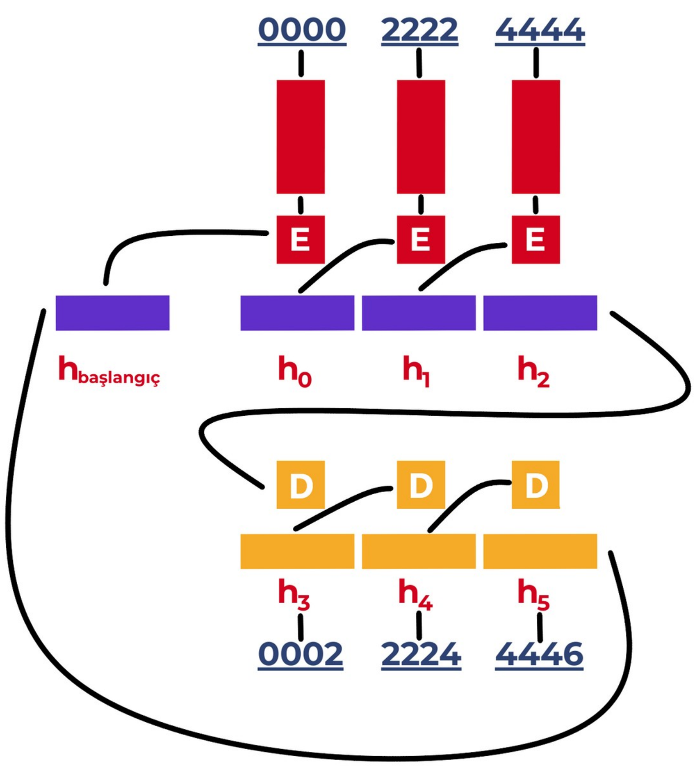
3 Kodlayıcı ve Kod Çözücü Yığınları

Kodlayıcı: Kodlayıcı, N = 6 özdeş katman yığınından oluşur. Her katmanın iki alt katmanı vardır. İlki çok kafalı bir kendi kendine dikkat mekanizmasıdır ve ikincisi basit, konumsal olarak tam bağlantılı bir ileri besleme ağıdır. Her alt katmanın çıktısı (x + Alt Katman (x)) olarak gösterilebilir; burada Alt Katman (x), alt katmanın kendisi tarafından uygulanan işlevdir. Bu artık bağlantıları kolaylaştırmak için, modeldeki tüm alt katmanların yanı sıra gömme katmanları d_{model} = 512 boyutunda çıktılar üretir.

Kod Çözücü: Kod çözücü de bir N = 6 özdeş katman yığınından oluşur. Her kodlayıcı katmanındaki iki alt katmana ek olarak, kod çözücü, kodlayıcı yığınının çıktısı üzerinde çok başlı dikkat çeken üçüncü bir alt katman ekler. Ayrıca, konumların sonraki konumlara katılmasını önlemek için kod çözücü yığınındaki öz dikkat alt katmanını da değiştirilmelidir. Bu maskeleme, çıktı düğümlerinin bir konumla dengelenmesi gerçeğiyle birleştiğinde, i konumu için tahminlerin sadece i'den daha düşük konumlarda bilinen çıkışlara bağlı olabilmesini sağlar.

4 Yeni Değer Üretimi

Adam Ive Transormatör modelli yeni değer üretimi için çok basit bir yol izler. Bu yolu Şekil 2 üzerinden gözlemleyebiliriz. Transformatörümüze üç adet girdi verilmiş olsun, bunlar sırası ile; 0000, 2222, 4444. $H_{başlangıç}$ kısmı ilk **E**ncoder'a (Kodlayıcı) bağlanıyor ve bu 0000 değeri olarak sisteme girilmiş olsun. Ardından H_0 ikinci sistem girdisi olan 2222'ye bağlanıyor. Son olarak H_1 üçüncü ve son girdi olan 4444'e bağlanılıyor. Transformatörümüzün kendi değerlerini oluşturduğu kısma geldik, kendinden önce bulunan girdileri göz önünde bulundurarak. H_2 ilk **D**ecoder'a (Kod Çözücü) bağlanıyor ve kod çözücümüz H_3'e 0002 değerini tanımlıyor. İlk üretimini gerçekleştiren transformatör aynı döngüyü tekrar tekrar tamamlayarak sistemin istediği türde müzikleri üretmeyi başarıyor. (Elbette her döngü başlangıcı bir önce üretilen değer örnek alınarak yapılıyor. Yani H_6, 4446 değerini örnek alarak tekrarlama işlemine devam edecektir.)



Şekil 2: Transformatör – çalışma prensibi

5 Sonuç

Bu çalışma ile, kodlayıcı - kod çözücü modelleri ve tekrarlayan katmanların yerini öz-dikkatle değiştirip yenileme sistemlerini kullanarak müzik üretebilen Adam Ive transformatör modelini sunduk. Adam Ive, yedi farklı enstrümanla Chopin, Bach, Mozart, Beatles vb. stilleri birleştirebilen ve bunlarla 4 dakikalık müzik besteleyebilen sinir ağı ile beslenen transformatördür. Adam Ive, yüz binlerce MIDI dosyasındaki bir sonraki değeri tahmin etmeyi öğrenerek uyum, ritim ve stilleri öğrendi. Bir sonraki ses veya metinleri tahmin etmek üzere eğitilmiş büyük ölçekli transformatörler ile aynı modeli kullanmaktadır. Adam Ive'i eğitmek ve geliştirmek için hazırladığımız sisteme https://github.com/mendaxyazilim/project-adam adresinde ulaşabilirsiniz.

Referanslar

[1] Ilya Sutskever, Oriol Vinyals, and Quoc VV Le. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems*, pages 3104–3112, 2014.

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473, 2014.

[3] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *CoRR*, abs/1406.1078, 2014.

[4] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin. Attention Is All You Need. arXiv:1706.03762